**ACRO**

v1.6b  2015/05/10

Typeset Acronyms

Clemens Niederberger

https://bitbucket.org/cgnieder/acro/

contact@mychemistry.eu

**ACRO** not only allows you to create acronyms in a simple way but also lets you add them to different classes of acronyms. Lists can be created of separate classes wherever you want the list to appear.

**ACRO** also provides an option `single` which ignores acronyms that are used only once in the whole document.

As an experimental feature **ACRO** also offers the option `sort` which automatically sorts the list created by `\printacronyms`.

Table of Contents

1. Licence and Requirements 2
2. About 2
3. News 3
   3.1. Version 1.6 3
4. Basics 3
   4.1. Creating New Acronyms 3
   4.2. Using the Acronyms – the Commands 6
   4.3. Plural Forms 8
   4.4. Alternative Short Forms 9
   4.5. Extra Information for the List Entry 9
   4.6. Foreign Language Acronyms 10
5. Additional Commands and Possibilities 11
   5.1. Indefinite Forms 11
6. Printing the List 15
7. Options and Customization 16
   7.1. General Options 16
   7.2. Options Regarding Acronyms 18
   7.3. Options Regarding the List 20
8. About Page Ranges 22
9. Language Support 22
1. Licence and Requirements

Permission is granted to copy, distribute and/or modify this software under the terms of the \LaTeX X Project Public License (\LPPL), version 1.3 or later (http://www.latex-project.org/lppl.txt). The software has the status “maintained.”

\acro loads and needs the following packages: expl3,\footnote{on CTAN as l3kernel: http://mirrors.ctan.org/macros/latex/contrib/l3kernel/} xparse, xtemplate, l3keys2e,\footnote{on CTAN as oberdiek: http://mirrors.ctan.org/macros/latex/contrib/oberdiek/} zref-abspage and translations\footnote{see http://tex.stackexchange.com/q/59840/5049 for solutions.} [Nie13].

2. About

Acronyms and initialisms are abbreviations formed from the initial components in a phrase or a word. These components may be individual letters (as in CEO) or parts of words (as in Benelux and Ameslan). There is no universal agreement on the precise definition of the various terms nor on written usage.\footnote{http://tex.stackexchange.com/q/59449/5049}

After \textsc{Wikipedia} told us what acronyms are and we won’t confuse them with units or other kinds of abbreviations – why would we need another package for them? There are several already: acronym [Oet12], acromake [Tho95], acroterm [Vos10], the abbreviations package abbrevs [Swi01] (the current version 1.4 has a bug,\footnote{Not in the sense of \texttt{\textbackslash meaning}!} though), the nomenclature package nomencl [Vey+05], and of course the mighty glossaries [Tal13]. So there is really no need for a new package.

On the other hand acronym, the best of the acronym specific packages, has one or two shortcomings and sometimes using glossaries seems a bit of an overkill (or simply inconvenient as one has to run makeglossaries, makeindex or xindy, then\footnote{Rumour has it there is going to be a version that can be used without running an external program}). So \acro stands somewhere in between (but closer to acronym).

The main reason for the existance of \acro is a question on \TeX sx which intrigued me and in consequence led to \acro and it’s option single.\footnote{on CTAN as l3packages: http://mirrors.ctan.org/macros/latex/contrib/l3packages/}

\acro has many similarities with the acronym package. In fact, quite some macros have the same name and meaning.\footnote{on CTAN as translations: http://mirrors.ctan.org/macros/latex/contrib/translations/}

Please take a minute to think and decide which package will suit your needs best. Are you planning to add a glossary to your book? You should probably go with glossaries, then.
you planning to add a nomenclature? You may want to choose nomencl (or again: glossaries) and so on. ACRO does a good job for lists of abbreviations.

3. News

3.1. Version 1.6

Support for versions 0.* has been dropped.

4. Basics

4.1. Creating New Acronyms

Acronyms are created with the command \DeclareAcronym that can only be used in the preamble.

\DeclareAcronym{⟨id⟩}{⟨list of keys⟩}

The basic command for declaring an acronym.

This command understands a number of keys which are listed below. Some of them are not described immediately but at appropriate places in the documentation.

- short = {⟨text⟩} (required)
  the short form of the acronym.

- long = {⟨text⟩} (required)
  the long form of the acronym.

- short-plural = {⟨text⟩}
  the plural ending appended to the short form.

- long-plural = {⟨text⟩}
  the plural ending appended to the long form.

- long-plural-form = {⟨text⟩}
  plural long form of the acronym; replaces the long form when used instead of appending the plural ending.

- list = {⟨text⟩}
  if specified this will be written in the list as description instead of the long form.

- short-indefinite = {⟨text⟩}
  indefinite article for the short form.

- long-indefinite = {⟨text⟩}
  indefinite article for the long form.


**4. Basics**

\[\texttt{long-pre} = \{\texttt{⟨text⟩}\}\]

\(<\texttt{⟨text⟩}\) is prepended to the long form in the text but not in the list of acronyms.

\[\texttt{long-post} = \{\texttt{⟨text⟩}\}\]

\(<\texttt{⟨text⟩}\) is appended to the long form in the text but not in the list of acronyms.

\[\texttt{alt} = \{\texttt{⟨text⟩}\}\]

alternative short form.

\[\texttt{alt-indefinite} = \{\texttt{⟨text⟩}\}\]

indefinite article for the alternative form. Default: a

\[\texttt{extra} = \{\texttt{⟨text⟩}\}\]

extra information to be added in the list of acronyms.

\[\texttt{foreign} = \{\texttt{⟨original long form⟩}\}\]

can be useful when dealing with acronyms in foreign languages, see section 4.6 for details.

\[\texttt{sort} = \{\texttt{⟨text⟩}\}\]

if used the acronym will be sorted according to this key instead of its ID.

\[\texttt{class} = \{\texttt{⟨text⟩}\}\]

the class the acronym belongs to.

\[\texttt{cite} = \{\langle\texttt{prenote}\rangle\}\{\langle\texttt{postnote}\rangle\}\{\langle\texttt{citation keys}\rangle\}\}

a citation that is printed to the acronym according to an option explained later.

\[\texttt{short-format} = \{\langle\texttt{TEX code}\rangle\}\]

the format used for the short form of the acronym.

\[\texttt{long-format} = \{\langle\texttt{TEX code}\rangle\}\]

the format used for the long form of the acronym.

\[\texttt{first-long-format} = \{\langle\texttt{TEX code}\rangle\}\]

the format used for the first long form of the acronym as set with \texttt{\ac}, \texttt{\acf} or \texttt{\acflike} and their uppercase, plural and indefinite forms.

\[\texttt{pdfstring} = \{\langle\texttt{⟨text⟩/⟨plural ending⟩}\rangle\}\]

used as PDF string replacement in bookmarks when used together with the hyperref package. The appended plural ending is optional. If you leave it (and the /) the default ending is used.

\[\texttt{acccsupp} = \{\langle\texttt{⟨text⟩}\rangle\}\]

sets the ActualText key as presented by the accsupp package for the acronym.

\[\texttt{index-sort} = \{\langle\texttt{⟨text⟩}\rangle\}\]

If you use the package option index every occurrence of an acronym is recorded to the index and sorted by its ID or (if set) by the value of the sort key. This key allows to set an individual sorting option for the index. See section 5.7 for details.
4. Basics

\textbf{index} = \{(\text{text})\}

This key allows to overwrite the automatic index entry with an arbitrary one. See section 5.7 for details.

\textbf{index-cmd} = \{(\text{text})\}

This key let’s you set an individual index creating command for this acronym. It should be a command that takes one mandatory argument. See section 5.7 for details.

In its simplest form an acronym needs a short and a long form. Please note that both keys \textit{must} be set and that the \texttt{short} = \{(\text{k})\}ey \textit{must} always be the \textit{first} key that is set.

\begin{verbatim}
\% preamble:
\DeclareAcronym{test}{
  short = ST ,
  long = Some Test
}
\end{verbatim}

This creates the acronym “ST” with the ID “test” and the long form “Some Test.”

The \texttt{format} key allows you to choose a specific format for the short form of an acronym:

\begin{verbatim}
\% preamble:
\DeclareAcronym{ot}{
  short = ot ,
  long = Other Test ,
  short-format = \scshape
}
\end{verbatim}

The short form now looks like this: \texttt{OT}.

The \texttt{cite} key needs a bit explaining. It expects arguments like the standard \texttt{\cite} command, \textit{i.e.}, two optional arguments setting the \texttt{(prenote)} and \texttt{(postnote)} and one mandatory argument setting the citation key.

\begin{verbatim}
\% preamble:
\DeclareAcronym{ny}{
  short = NY ,
  short-plural = ,
  long = New York ,
  long-plural = ,
  cite = \{NewYork\}
}
\end{verbatim}
% bib file for use with biber/bibtex:
@online{NewYork,
  author = {Wikipedia},
  title = {New York City},
  urldate = {2012-09-27},
  url = {http://en.wikipedia.org/wiki/New_York_City},
  year = {2012}
}

The first appearance now looks as follows:\footnote{New York (NY) [Wik12b].}

\section*{4. Basics}

\subsection*{4.2. Using the Acronyms – the Commands}

Acronyms are used with one of the following commands:

\begin{itemize}
  \item \texttt{\textbackslash ac*\{\textit{id}\}}
    \begin{itemize}
      \item basic command; the first output is different from subsequent ones.
    \end{itemize}
  \item \texttt{\textbackslash Ac*\{\textit{id}\}}
    \begin{itemize}
      \item same as \texttt{ac} but capitalizes the first letter of the long form.
    \end{itemize}
  \item \texttt{\textbackslash acs*\{\textit{id}\}}
    \begin{itemize}
      \item short form; the actual acronym.
    \end{itemize}
  \item \texttt{\textbackslash acl*\{\textit{id}\}}
    \begin{itemize}
      \item long form; the meaning of the acronym.
    \end{itemize}
  \item \texttt{\textbackslash AcI*\{\textit{id}\}}
    \begin{itemize}
      \item same as \texttt{acl} but capitalizes first letter.
    \end{itemize}
  \item \texttt{\textbackslash aca*\{\textit{id}\}}
    \begin{itemize}
      \item alternative short form as specified in the \texttt{alt} key of \texttt{DeclareAcronym}; if it hasn’t been specified this is identical to \texttt{acs}.
    \end{itemize}
  \item \texttt{\textbackslash acf*\{\textit{id}\}}
    \begin{itemize}
      \item first form; output like the first time \texttt{ac} is output.
    \end{itemize}
  \item \texttt{\textbackslash Acf*\{\textit{id}\}}
    \begin{itemize}
      \item same as \texttt{acf} but capitalizes first letter of the long form.
    \end{itemize}
\end{itemize}

\footnote{The appearance of the citation of course depends on the citation style you’re using.}
4. Basics

\texttt{\textbackslash acp*\{\{id\}\}}

plural form of \texttt{\ac};

\texttt{\textbackslash Acp*\{\{id\}\}}

same as \texttt{\acp} but capitalizes first letter of the long form.

\texttt{\textbackslash acsp*\{\{id\}\}}

plural form of \texttt{\acs};

\texttt{\textbackslash aclp*\{\{id\}\}}

plural form of \texttt{\acl};

\texttt{\textbackslash Aclp*\{\{id\}\}}

same as \texttt{\aclp} but capitalizes first letter.

\texttt{\textbackslash acap*\{\{id\}\}}

plural form of \texttt{\aca};

\texttt{\textbackslash acfp*\{\{id\}\}}

plural form of \texttt{\acf};

\texttt{\textbackslash Acfp*\{\{id\}\}}

same as \texttt{\acfp} but capitalizes first letter of the long form.

If an acronym is used the first time with \texttt{\ac} its output is different from subsequent uses. To be clear on this: the first time! If the acronym has been used with any of the output commands before it is not the first time any more.

If you use the starred variant an acronym will not be marked as used. This proves useful if an acronym is typeset in a section title, for example, since then the appearance in the table of contents won’t mark it as used.

\begin{verbatim}
% preamble:
% \DeclareAcronym{cd}{
%   short = cd , long = Compact Disc ,
%   short-format = \scshape
%   first time: \ac{cd} \\ second time: \ac{cd} \\ alternative: \aca{cd} \\ first again: \acf{cd} \\ long: \acl{cd} \\ short plural: \acsp{cd} \\ long plural: \aclp{cd}
% }

\end{verbatim}

first time: Compact Disc (\textit{c\,d})
second time: \textit{c\,d}
short: \textit{c\,d}
alternative: \textit{c\,d}
first again: Compact Disc (\textit{c\,d})
long: Compact Disc
short plural: \textit{c\,d}s
long plural: Compact Discs
4. Basics

4.3. Plural Forms

If an acronym is defined in the standard way \acro uses an 's' that’s appended to both the short and the long form when one of the plural commands is used. However, that is not always the best solution. For one thing not all acronyms may have a plural form. Second, the plural form especially of the long forms may be formed differently. And third, other languages can have other plural endings.

For these reasons \DeclareAcronym can get the following keys:

- **short-plural** = \{⟨text⟩\} 
  - Default: s
  - The plural ending of the short form.

- **long-plural** = \{⟨text⟩\} 
  - Default: s
  - The plural ending of the long form.

- **long-plural-form** = \{⟨text⟩\}
  - An alternative plural form for the long form.

These keys are optional. If they’re not used, the default setting is s. If you use **long-plural-form** the long form will be replaced by the specified plural form when necessary.

Suppose we define the following acronyms:

```latex
\DeclareAcronym{cd}{
  short = cd ,
  long = Compact Disc ,
  short-format = \textscshape
}
\DeclareAcronym{ny}{
  short = NY ,
  short-plural = ,
  long = New York ,
  long-plural =
}
\DeclareAcronym{sw}{
  short = SW ,
  long = Sammelwerk ,
  long-plural = e
}
\DeclareAcronym{MP}{
  short = MP ,
  long = Member of Parliament ,
  long-plural-form = Members of Parliament
}
```

These acronyms now have the following plural appearances:
4. Basics

\(\texttt{\acs{cd}}, \texttt{\acl{cd}} \parallel\) Compact Discs
\(\texttt{\acs{ny}}, \texttt{\acl{ny}} \parallel\) New York
\(\texttt{\acs{sw}}, \texttt{\acl{sw}} \parallel\) Sammelwerke
\(\texttt{\acs{MP}}, \texttt{\acl{MP}} \parallel\) Members of Parliament

4.4. Alternative Short Forms

For some acronyms it might be useful to have alternative forms. For this \texttt{\DeclareAcronym} has another key:

\texttt{alt = \{\langle\text{text}\rangle\}}

Alternative short form.

\begin{verbatim}
% preamble:
% \DeclareAcronym{jpg}{
%  short = JPEG ,
%  alt = JPG ,
%  long = Joint Photographic Experts Group
% }
% default: \acs{jpg} \parallel
% alt.: \aca{jpg}

default: JPEG
alt.: JPG
\end{verbatim}

The alternative form uses the same plural ending as the default short form and is formatted in the same way.

4.5. Extra Information for the List Entry

Of course you can print a list of acronyms where their meaning is explained. Sometimes it can be useful to add additional information there. This is done with another key to \texttt{\DeclareAcronym}:

\texttt{extra = \{\langle\text{text}\rangle\}}

Additional information for the list of acronyms.

These information will only be displayed in the list. See section 6 for the impact of the following example.
4. Basics

The \ac{nato} is an intergovernmental military alliance based on the North Atlantic Treaty which was signed on 4~April 1949. \ac{nato} headquarters are in Brussels, Belgium, one of the 28 member states across North America and Europe, the newest of which, Albania and Croatia, joined in April 2009.

4.6. Foreign Language Acronyms

I repeatedly read the wish for being able to add translations to acronyms when the acronyms stem from another language than the document language, \textit{i.e.}, something like the following in a German document:

\begin{verbatim}
\ac{ecu}\\
\ac{ecu}
\end{verbatim}

Steuergerät (Electronic Control Unit, ECU)

ECU

That’s why I decided to add the following key:

\texttt{foreign = \{\text{original long form}\}}

A description for an acronym originating in another language than the document language.

Here is the definition of the above mentioned ECU acronym:

\begin{verbatim}
\DeclareAcronym{ecu}{
  short = ECU ,
  long = Steuergerät ,
  foreign = Electronic Control Unit
}
\end{verbatim}
As you have seen this adds the foreign entry to the first appearance of an acronym. It is also added in parentheses to the list of acronyms after the long entry. Actually the entry there is the argument to the following command:

\acroenparen{(argument)}

Places (argument) in parentheses: \acroenparen{example}: (example). See page 19 for a way to customize this other than redefining it.

5. Additional Commands and Possibilities

5.1. Indefinite Forms

Unlike many other languages in English the indefinite article is not determined by the grammatical case, gender or number but by the pronunciation of the following word. This means that the short and the long form of an acronym can have different indefinite articles. For these cases \acro offers the keys short-indefinite, alt-indefinite and long-indefinite whose default is a. For every lowercase singular command two alternatives exist, preceded by i and I, respectively, which output the lowercase and uppercase version of the corresponding indefinite article.

\% preamble:
\% \DeclareAcronym{ufo}{
  \% short = UFO ,
  \% long = unidentified flying object ,
  \% long-indefinite = an
\% }
\iac{ufo}; \iacs{ufo}; \iacl{ufo}

An Unidentified flying object (UFO); a UFO; an unidentified flying object

5.2. Uppercasing

\acfirstupper{(token list)}

This command uppercases the first token in (token list). The command is less powerful than \makefirstuc that is provided by the mfirstuc package [Tal12] but it is expandable. Obvious downsides are for example that it does not uppercase accented letters.

5.3. Simulating the First Appearance

Users told me that there are cases when it might be useful to have the the acronym typeset according to the first-style but with another text than the long form. For such cases \acro

---

10. Let’s better say: unlike the other languages where I know at least the basics.
11. Well – one, to be precise :)
5. Additional Commands and Possibilities

offers the following commands.
\acro{\{id\}}{(instead of long form)}
Write some alternative long form for acronym \textit{id} as if it were the first time the acronym was used.
\acropl{\{id\}}{(instead of long form)}
Plural form of \acro{\{id\}}.

\begin{itemize}
\item \acs{ny} (NY)
\item \acs{la} (Los Angeles)
\end{itemize}

\begin{itemize}
\item NY\(^a\)
\item a the big apple [Wik12b]
\end{itemize}

The plural ending in \acropl{\{id\}} is only appended to the short form. It makes no sense to append it to the text that is inserted manually anyway. Note that whatever text you’re inserting might be gobbled depending on the \textit{first-style} you’re using.

### 5.4. Using Classes

The acronyms of \texttt{ACRO} can be divided into different classes. This doesn’t change the output but allows different acronym lists, see section 6. For this \texttt{DeclareAcronym} has an additional key:

\begin{verbatim}
\DeclareAcronym{\{id\}}{\{text\}}
\end{verbatim}

Associated class for an acronym.

This might be useful if you can and want to divide your acronyms into different types, technical and grammatical ones, say, that shall be listed in different lists.

\begin{verbatim}
% preamble:
\% \DeclareAcronym{la}{
\% short = LA ,
\% short-plural = ,
\% long = Los Angeles ,
\% long-plural = ,
\% class = city
\%}
\% \DeclareAcronym{ny}{
\% short = NY ,
\% short-plural = ,
\% long = New York ,
\% long-plural = ,
\% class = city
\% cite = NewYork
\%}
\acl{la} (\acs{la}) \acl{ny} (\acs{ny})
\end{verbatim}

Los Angeles (LA)
New York (NY)
5. Additional Commands and Possibilities

5.5. Reset or Mark as Used, Test if Acronym Has Been Used

If you want for some reason to fool \acro into thinking that an acronym is used for the first time you can call one of these commands:

\acroreset{⟨comma separated list of ids⟩}

This will reset a used acronym such that the next use of \ac will again print it as if it were used the first time. This will not remove an acronym from being printed in the list if it actually has been used before.

\acroresetall

Reset all acronyms.

\acifused{⟨id⟩}{⟨true⟩}{⟨false⟩}

This command tests if the acronym with ID ⟨id⟩ has already been used and either puts true or false in the input stream.

\acroreset{ny} \ac{ny}

New York (NY) [Wik12b]

Beware that both commands act globally! There are also commands that effectively do the opposite of \acroreset, i.e., mark acronyms as used:

\acuse{⟨comma separated list of ids⟩}

This has the same effect as if an acronym had been used twice, that is, further uses of \ac will print the short form and the acronym will in any case be printed in the list (as long as its class is not excluded).

\acuseall

Mark all acronyms as used.

5.6. \ac and Friends in PDF Bookmarks

\acro’s commands usually are not expandable which means they’d leave unallowed tokens in PDF bookmarks. hyperref offers \texorpdfstring to circumvent that issue manually but that isn’t really a nice solution. What’s the point of having macros to get output for you if you have to specify it manually after all? That is why \acro offers a preliminary solution for this. In a bookmark every \ac like command falls back to a simple text string typesetting what \acs would do (or \acsp for plural forms). These text strings both can accessed manually and can be modified to an output reserved for PDF bookmarks.

\acpdfstring{⟨id⟩}

Access the text string used in PDF bookmarks.
\acpdfstringplural{(id)}

Access the plural form of the text string used in PDF bookmarks.

pdfstring = {{\langle pdfstring\rangle}/{\langle plural ending\rangle}}

Key for \DeclareAcronym to declare a custom text string for PDF bookmarks. The plural ending can be set optionally.

accsupp = {{\langle text\rangle}}

Key for \DeclareAcronym to set the ActualText property of \BeginAccSupp (see accsupp's documentation for details) to be used for an acronym. It only has an effect when the package option accsupp is used, too.

For example the PDF acronym used in the title for this section is defined as follows:

\begin{verbatim}
\DeclareAcronym{pdf}{
    short = pdf ,
    long = Portable Document Format ,
    format = \scshape ,
    pdfstring = PDF ,
    accsupp = PDF
}
\end{verbatim}

This also demonstrates the accsupp key. For this to work you need to use the package option accsupp, too, which will load the package accsupp. Then the key accsupp will set the ActualText property of \BeginAccSupp. Please refer to accsupp's documentation for details. To see its effect copy PDF and paste it into a text file. You should get uppercase letters instead of lowercase ones.

5.7. Adding Acronyms to the Index

ACRO has the package option index. If it is used an index entry will be recorded every time an unstarrred acronym command is used. The index entry will be \langle id\rangle@\langle short\rangle, \langle sort\rangle@\langle short\rangle if the sort key has been set, \langle index-sort\rangle@\langle short\rangle if the index-sort has been set, or \langle index\rangle if the key index has been set for the specific acronym. The short versions appearing there are formatted according to the chosen format of the corresponding acronym, of course.

This document demonstrates the feature. You can find every acronym that has been declared in the index. In order to allow flexibility the indexing command can be chosen both globally via package option and individually for every acronym. This would allow to add acronyms to a specific index if more than one index is used, for example with help of the imakeidx package.

I'm not yet convinced this is a feature many people if anyone needs and if they do if it is flexible enough. If you have any thoughts on this I'd appreciate an email.
6. Printing the List

Printing the whole list of acronyms is easy: just place `\printacronyms` wherever you want the list to be.

\printacronyms[⟨options⟩]

Print the list of acronyms.

The commands takes a few options, namely the following ones:

include-classes = {⟨list of classes⟩}

Takes a comma-separated list of the classes of acronyms that should be in the list.

exclude-classes = {⟨list of classes⟩}

Takes a comma-separated list of the classes of acronyms that should not be in the list.

name = {⟨name of the list⟩}

Sets the name for the list.

heading = {⟨sectioning command without leading backslash⟩}  

Default: section*

Sets the sectioning command for the heading of the list. A special value is none which suppresses the heading.

sort = true|false

Default: true

Set sorting for this list only.

\acsetup{extra-style=comma}
\printacronyms[exclude-classes=city]
\printacronyms[include-classes=city,name={City Acronyms}]

### Acronyms

- **CD** Compact Disc
- **CTAN** Comprehensive \TeX{} Archive Network
- **ECU** Steuergerät (Electronic Control Unit)
- **ID** identification string
- **JPEG** Joint Photographic Experts Group
- **MP** Member of Parliament
7. Options and Customization

**NATO** North Atlantic Treaty Organization, *deutsch*: Organisation des Nordatlantikvertrags

**PDF** Portable Document Format

**SW** Sammelwerk

**ST** Some Test

**UFO** unidentified flying object

**City Acronyms**

**LA** Los Angeles

**NY** New York

You can see that the default layout is a description list with a \section title. Both can be changed, see section 7. The command \printacronyms needs two \LaTeX runs. This is a precaution to avoid error messages with a possibly empty list. But since almost all documents need at least two runs and often are compiled much more often than that, this fact shouldn’t cause too much inconvenience.

7. Options and Customization

7.1. General Options

There are a few options which change the general behaviour of \acro. Underlined values are used if no value is given.

**messages = silent|loud**

Default: loud

Setting messages = {silent} will turn all of \acro’s error messages into warnings and all of \acro’s warnings into info messages. Be sure to check the log file carefully if you decide to set this option.

**single = true|false**

Default: false

If set to true an acronym that’s used only once (with \ac) in a document will only print the long form and will not be printed in the list.

**hyperref = true|false**

Default: false

If set to true the short forms of the acronyms will be linked to their list entry.

**label = true|false**

Default: false

If set to true this option will place \label{⟨prefix⟩⟨id⟩} the first time the acronym with \id ⟨id⟩ is used.

**label-prefix = {{text}}**

Default: ac:

The prefix for the \label that is placed when option label = {true} is used.
### Options and Customization

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>record-pages</code></td>
<td>true/false</td>
<td>Since <code>acro</code> can handle Arabic, Roman and Roman page numbers but not any other kind of numbering, this option allows to turn the page number recording off for these cases as it would lead to errors else. This affects the whole document and can only be set in the preamble! It means you cannot have page numbers in the list of acronyms in this case. Or rather: you can if you use <code>pages = {first}</code>.</td>
</tr>
<tr>
<td><code>only-used</code></td>
<td>true/false</td>
<td>This option is true as default. It means that only acronyms that are actually used in the document are printed in the list. If false, all acronyms defined with \DeclareAcronym will be written to the list.</td>
</tr>
<tr>
<td><code>mark-as-used</code></td>
<td>first/any</td>
<td>This option determines whether an acronym is marked as used when the first form is used the first time (with \ac, \acf or \acfl like and their uppercase, plural and indefinite forms) or when any of the \ac-like commands is used.</td>
</tr>
<tr>
<td><code>macros</code></td>
<td>true/false</td>
<td>If set to true this option will create a macro \langle id\rangle for each acronym as a shortcut for \ac{⟨id⟩}. Already existing macros will not be overwritten.</td>
</tr>
<tr>
<td><code>xspace</code></td>
<td>true/false</td>
<td>If set to true this option will append \xspace from the xspace package to the commands created with the macros option.</td>
</tr>
<tr>
<td><code>strict</code></td>
<td>true/false</td>
<td>If set to true and the option macros = {true} is in effect then already existing macros will be overwritten.</td>
</tr>
<tr>
<td><code>sort</code></td>
<td>true/false</td>
<td>If set to true the acronym list will be sorted automatically. The entries are sorted by their id ignoring upper and lower case. This option needs the experimental package l3sort (from the l3experimental bundle) and can only be set in the preamble.</td>
</tr>
<tr>
<td><code>cite</code></td>
<td>all/first/none</td>
<td>This option decides whether citations that are added via cite are added to each first, every or no appearance of an acronym.</td>
</tr>
<tr>
<td><code>cite-cmd</code></td>
<td>\cite</td>
<td>This option determines which command is used for the citation. Each citation command that takes the cite key as argument is valid, for example biblatex's \footcite.</td>
</tr>
<tr>
<td><code>cite-space</code></td>
<td>\nobreakspace</td>
<td>Depending on the citation command in use a space should be inserted before the citation or maybe not (e.g. \footcite...). This option allows you to set this. Actually it can be used to place arbitrary code right before the citation.</td>
</tr>
</tbody>
</table>
7. Options and Customization

index = true|false
Default: false
If set to true an index entry will be recorded every time an unstarred acronym command is used for the corresponding acronym.

index-cmd = \{⟨control sequence⟩\}
Default: \index
Chooses the index command that is used when option index has been set to true.

accsupp = true|false
Default: false
Activated the access support as provided by the accsupp package.

uc-cmd = \{⟨control sequence⟩\}
Default: \acfirstupper
The command that is used to capitalize the first word in the \Ac and the like commands. You can change it to another one like for example \makefirstuc\textsuperscript{12} or \MakeTextUppercase\textsuperscript{13}.

All options of this and the following sections can be set up either as package options or via the setup command:

\acsetup\{⟨options⟩\}
Set up ACRO anywhere in the document. Or separate package loading from setup.

\begin{quote}
% with \acsetup\{macros\}
we could have used these before: \texttt{nato, ny}
\end{quote}

we could have used these before: Nato, NY

7.2. Options Regarding Acronyms

The options described in this section all influence the layout of one of the possible output forms of the acronyms.

short-format = \{⟨format⟩\}
(initially empty)
Sets a format for all short forms. For example short-format = \{scshape\} would print all short forms in small caps.

long-format = \{⟨format⟩\}
The same for the long forms.

foreign-format = \{⟨format⟩\}
(initially empty)
The format for the foreign entry when it appears as part of the first appearance of an acronym.

\begin{thebibliography}{9}
\bibitem[\textsuperscript{12}]{12}
\textsuperscript{12} from the mfirstuc package
\textsuperscript{13}
\textsuperscript{13} from the textcase package
\end{thebibliography}
7. Options and Customization

**first-long-format** = \{\langle format\rangle\} (initially empty)

The format for the long form on first usage (with \texttt{ac}, \texttt{acf} or \texttt{acflke} and their uppercase, plural and indefinite forms).

**list-short-format** = \{\langle format\rangle\} (initially empty)

An extra format for the short entries in the list. If not used this is the same as \texttt{short-format}. Please be aware that a call of \texttt{short-format} after this one will overwrite it again.

**list-long-format** = \{\langle format\rangle\} (initially empty)

An extra format for the long entries in the list. If not used this is the same as \texttt{long-format}. Please be aware that a call of \texttt{long-format} after this one will overwrite it again.

**list-foreign-format** = \{\langle format\rangle\}

Default: \texttt{acroenparen}

The format for the foreign entry as it appears in the list. This may be code that ends with a macro that takes a mandatory argument.

**extra-format** = \{\langle format\rangle\} (initially empty)

The same for the extra information.

**first-style** = default|plain|empty|square|short|reversed|plain-reversed|footnote|sidenote

Default: default

Changed in version 1.1

The basic style of the first appearance of an acronym. The value sidenote needs the command \texttt{sidenote} to be defined for example by the sidenotes package.

**extra-style** = default|plain|comma|paren|bracket

Default: default

Defines the way the extra information is printed in the list.

**plural-ending** = \{\langle tokenlist\rangle\}

Default: s

With this option the default plural ending can be set.

\begin{verbatim}
% (Keep in mind that we're in a minipage here!)
\acrosetup{first-style=empty}
\acf{ny}  \hfill NY
\acrosetup{first-style=footnote}
\acrosetup{first-style=square}
\acf{ny}  \hfill NY \[Wik12b\]
\acrosetup{first-style=short}
\acf{ny}  \hfill New York [NY]
\acrosetup{first-style=reversed}
\acf{ny}  \hfill NY
\acrosetup{first-style=plain}
\acf{ny}  \hfill NY
\acrosetup{first-style=plain-reversed}
\acf{ny}
\end{verbatim}
7. Options and Customization

7.3. Options Regarding the List

page-ref = none/plain/comma/paren  
Default: none
If this option is set to a value other than none the page numbers of the an acronym appeared on are printed in the list. Please note that this is an experimental feature and might fail in quite a number of cases. If you notice anything please send me an email!

pages = all/first  
Default: all
If the option page-ref has any value other than none this option determines wether all usages of the acronyms are listed or only the first time. Implicitly sets label = {true}.

page-name = {⟨page name⟩}  
Default: p.@@
The “name” of the page label. This is automatically translated to the active language. However for the time being there are many translations missing, yet. Please notify me if you find your language missing.

pages-name = {⟨page name plural⟩}  
Default: pp.@@
The “name” of the page label when there are more than one page. This is automatically translated to the active language. However for the time being there are many translations missing, yet. Please notify me if you find your language missing.

following-page = true/false  
Default: false
If set to true a page range in the list of acronyms that consists of two pages will be written by the first page and an appended f. This depends on the option next-page.

following-pages = true/false  
Default: false
If set to true a page range in the list of acronyms that set consists of more than two pages will be written by the first page and an appended ff. This depends on the option next-pages.

next-page = {⟨text⟩}  
Default: \,f.@
Appended to a page number when following-page is set to true and the range is only 2 pages long. This is automatically translated to the active language. However, for the time being there are many translations missing, yet. Please notify me if you find your language missing.

next-pages = {⟨text⟩}  
Default: \,ff.@
Appended to a page number when following-pages is set to true and the range is more than 2 pages long. This is automatically translated to the active language. However, for the time being there are many translations missing, yet. Please notify me if you find your language missing.

list-type = table|⟨list⟩  
Default: description
This option let’s you choose how the list is printed. ⟨list⟩ can be any valid list like itemize or description.

list-style = list|tabular|longtable|extra-tabular|extra-longtable|extra-tabular-rev|extra-longtable-rev  
Default: list
If you choose list-type = {table} you have to specify which kind of table should be used. If you choose longtable, extra-longtable or extra-longtable-rev you have to load longtable
in your preamble. The values `extra-longtable` and `extra-longtable-rev` put the extra information in a column of its own.

**list-heading** = chapter|chapter*|section|section*|subsection|subsection*|addchap|addsec|none

Default: `section*`

The heading type of the list. The last two only work with a KOMA-Script class that also defines the appropriate command. A special value is `none` which suppresses the heading.

**list-name** = `{⟨list name⟩}`

Default: `Acronyms`

The name of the list. This is what’s written in the list-heading. This is automatically translated to the active language. However, for the time being there are many translations missing, yet. Please notify me if you find your language missing.

**list-table-width** = `{⟨dimension⟩}`

Default: `.7\linewidth`

This has only an effect if you chose `list-type` = `{table}`. The second column (or the third if you choose one of the `-rev` styles) of the table is a `p` column whose width can be specified with this option.

**list-caps** = `true`|`false`

Default: `false`

Print the first letters of the long form capitalized.

If you for example have loaded `enumitem` you can define a custom list for the acronym list:

```latex
\% preamble:
\% \usepackage{enumitem}
\newlist{acronyms}{description}{1}
\newcommand*{\addcolon}[1]{#1:}
\setlist[acronyms]{
  labelwidth=3em,
  leftmargin=3.5em,
  noitemsep,
  itemindent=0pt,
  font=\addcolon}
\acsetup{list-type=acronyms,hyperref=false,extra-style=comma}
\printacronyms
```

### Acronyms

- **CD**: Compact Disc
- **CTAN**: Comprehensive \TeX\ Archive Network
- **ECU**: Steuergerät (Electronic Control Unit)
- **ID**: identification string
8. About Page Ranges

If you enable the page-ref option \acro adds page numbers to the list of acronyms. In version 0.x it would add a page reference for an acronym in the list of acronyms that used \pageref to refer to the first appearance of an acronym. This is retained using `pages = {first}`. Version 1.0 uses a different approach that doesn’t use a label but instead will list all pages an acronym appeared on. With hyperref the pages are referenced using `\hyperpage`.

There are some options that control how this list will be typeset, e.g., `following-page`, `next-pages` or the option `page-ref` itself. It is important to mention that the page list will always take at least two compilation runs until changes in the options or the actual page numbers affect it. This is due to the fact that the updated sequence is first written to the aux file and only read in during the next run.

9. Language Support

\acro detects if packages babel or polyglossia are being loaded and tries to adapt certain strings to match the chosen language. However, due to my limited language knowledge only a few translations are provided. I’ll show how the English translations are defined so you can add the translations to your preamble if needed. Even better would be you’d send me a short email at contact@mychemistry.eu with the appropriate translations for your language and I’ll add them to \acro.

```latex
1 \DeclareTranslation{English}{acronym-list-name}{Acronyms}
2 \DeclareTranslation{English}{acronym-page-name}{p.}
3 \DeclareTranslation{English}{acronym-pages-name}{pp.}
4 \DeclareTranslation{English}{acronym-next-page}{f.}
5 \DeclareTranslation{English}{acronym-next-pages}{ff.}
```
10. hyperref Support

The option \hyperref = \{true\} adds internal links from all short (or alternative) forms to their respective list entries. Of course this only works if you have loaded the \hyperref package in your preamble. You should use this option with care: if you don’t use \printacronyms anywhere this option will result in loads of \hyperref warnings. Also printing several lists can result in warnings if don’t clearly separate the lists into different classes. If an acronym appears in more than one list there will also be more than one hypertarget for this acronym.

Using \hyperref will also add \hyperpage to the page numbers in the list (provided they are displayed in the style chosen). Like with an index the references will thus not point to the acronyms directly but to the page they’re on.

A. All Acronyms Used in this Documentation

\begin{verbatim}
\acsetup{
  list-type = table,
  list-style = longtable,
  list-heading = subsection*,
  extra-style = comma,
  page-ref = comma
}
\printacronyms[name=All Acronyms]
\end{verbatim}

All Acronyms

\begin{description}
\item[CD] Compact Disc, pp. 7, 9
\item[CTAN] Comprehensive \TeX\ Archive Network, p. 2
\item[ECU] Steuergerät (Electronic Control Unit), p. 10
\item[ID] identification string, pp. 4, 5, 13, 16, 17
\item[JPEG] Joint Photographic Experts Group, p. 9
\item[LA] Los Angeles, p. 12
\item[MP] Member of Parliament, p. 9
\item[NATO] North Atlantic Treaty Organization, \textit{deutsch}: Organisation des Nordatlantikvertrags, pp. 10, 18
\item[NY] New York, pp. 6, 9, 12, 13, 18, 19
\item[PDF] Portable Document Format, pp. 4, 13, 14
\item[SW] Sammelwerk, p. 9
\item[ST] Some Test, p. 5
\item[UFO] unidentified flying object, p. 11
\end{description}
B. References

City Acronyms

<table>
<thead>
<tr>
<th>City</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LA</td>
<td>Los Angeles, p. 12</td>
</tr>
<tr>
<td>NY</td>
<td>New York, pp. 6, 9, 12, 13, 18, 19</td>
</tr>
</tbody>
</table>

B. References


C. Index

A
abbrevs (package) ........................................... 2
\Ac ......................................................... 6, 18
\ac ...................................................... 4, 6 f., 10, 13, 16 f., 19
\aca ....................................................... 6 f., 9
\acap ......................................................... 7
accsupp (package) ........................................... 4, 14, 18
\Acf ......................................................... 4, 14, 18
\acf ......................................................... 4, 6 f., 17, 19
\acfirstupper ................................................ 11, 18
\acflike ..................................................... 4, 12, 17, 19
\Acfp ......................................................... 7
\acfp ......................................................... 7
\acfplike ................................................... 12
\acifused .................................................... 13
\Ac ......................................................... 6
\acl ......................................................... 6 f., 12
\Accl ......................................................... 7
\accl ......................................................... 7, 9
\Ac ......................................................... 7
\Ac ......................................................... 7
\acpfstring .................................................. 13
\acpfstringplural .......................................... 14
\acreset ..................................................... 13
\acresetall .................................................. 13
\acroparen .................................................. 11, 19
acromake (package) ........................................... 2
acronym (package) ......................................... 2
\Acronym and initialism .................................... 2
acroterm (package) ......................................... 2
\acs ....................................................... 6 f., 9, 12 f.
\acssetup .................................................... 12, 15, 18 f., 21, 23 f.
\acs ....................................................... 7, 9, 13
\acuse ...................................................... 13
\acuseall ................................................... 13
alt ......................................................... 4, 6, 9
alt-indefinite ............................................... 4, 11

C
cD ......................................................... 7, 9
cite ....................................................... 4 f., 17
cite-cmd ..................................................... 17
cite-space .................................................. 17
class ....................................................... 4, 12
CTAN ....................................................... 2
\DeclareAcronym ........................................... 3, 5 f., 8 f., 12, 14, 17
description (environment) ................................. 20
ECU ......................................................... 10
enumitem (package) ........................................ 21
exclude-classes ............................................ 15
expl3 (package) ........................................... 2
extra ...................................................... 4, 9
extra-format ............................................... 19
extra-style ............................................... 19

F
first-long-format ......................................... 4, 19
first-style ............................................... 11 f., 19
following-page .......................................... 20, 22
following-pages ........................................... 20
foreign ...................................................... 4, 10 f., 18 f.
foreign-format ............................................ 18
format .................................................... 5

G
glossaries (package) ....................................... 2 f.

H
heading ..................................................... 15
hyperref .................................................... 16, 23
hyperref (package) ....................................... 4, 13, 22 f.

I
\Iac ....................................................... 11
\Iacl ...................................................... 11
\Iacs ..................................................... 11
include-classes .......................................... 15
index ..................................................... 4 f., 14, 18
Index

index-cmd .................................. 5, 18
index-sort .................................. 4, 14
itemize (environment) ..................... 20

J
JPEG .............................................. 9

L
3experimental (package) .................. 17
3keys2e (package) ......................... 2
3sort (package) ............................... 17
LA .................................................. 12
label ............................................. 16, 20
label-prefix ................................... 16
list ............................................... 3
list-caps ........................................ 21
list-foreign-format ....................... 19
list-heading ................................... 21
list-long-format ............................. 19
list-name ....................................... 21
list-short-format ............................ 19
list-style ...................................... 20
list-table-width ............................. 21
list-type ....................................... 20f.
long ............................................ 3, 11
long-format ................................... 4, 18f.
long-indefinite ............................. 3, 11
long-plural ................................... 3, 8
long-plural-form ............................ 3, 8
long-post ....................................... 4
long-pre ........................................ 4
longtable (package) ....................... 20
LINDEX ............................................. 2

M
macros ........................................... 17
\makefirstuc .................................. 11
mark-as-used .................................. 17
messages ....................................... 16
mfirstuc (package) ......................... 11, 18
MP .................................................. 9

N
name ............................................. 15
NATO ............................................. 10, 18

New York City ............................... 6, 12f., 19
next-page ..................................... 20
next-pages ..................................... 20, 22
Niederberger, Clemens ..................... 2
nomencl (package) ......................... 2f.
NY ............................................... 6, 9, 12f., 18f.

O
Oetiker, Tobias ............................... 2
only-used ....................................... 17
OT ................................................. 5

P
page-name ..................................... 20
page-ref ....................................... 20, 22
pages ........................................... 17, 20, 22
pages-name ................................... 20
PDF ............................................... 4, 13f.
pdfstring ...................................... 4, 14
plural-ending .................................. 19
\printacronyms ............................... 1, 15f., 21, 23f.

R
record-pages .................................. 17

S
short ............................................ 3, 5
short-format .................................. 4, 18f.
short-indefinite ............................ 3, 11
short-plural ................................... 3, 8
single ........................................... 1f., 16
sort ................................................ 1, 4, 14f., 17
strict ............................................ 17
SW ............................................... 9
Swift, Matt ..................................... 2

T
Talbot, Nicola ............................... 2, 11
ST ................................................ 5
\TeX{\textit{sx}} .................................. 2
Thompson, Paul .............................. 2
translations (package) ..................... 2

U
uc-cmd ......................................... 18
UFO ............................................. 11
## Index

<table>
<thead>
<tr>
<th>V</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>Veytsman, Boris</td>
<td>xparse (package)</td>
</tr>
<tr>
<td>Voss, Jakob</td>
<td>xspace</td>
</tr>
<tr>
<td>Wikipedia</td>
<td>xtemplate (package)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>W</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wikipedia</td>
<td>zref-apsage (package)</td>
</tr>
</tbody>
</table>